Ashenafi Wakjira

Predicting Voting Affiliation Using Machine Learning Algorithms

Helsinki Metropolia University of Applied Sciences

Degree

Degree Programme

Thesis

Date 25 April 2014

Helsinki Metropolia University of Applied Sciences

Author(s) Title Number of Pages Date	Ashenafi Wakjira Predicting voting Affiliation Using Machine Learning Algorithms 41 pages + 3 appendices 25 April 2014
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Embedded System Engineering
Instructor(s)	Jaakko Pitkänen, Senior Lecturer

Human beings are brave enough to read, understand and draw conclusions in many areas. When it comes to complex, multidimensional, junky data, decision making becomes difficult, time consuming, erroneous and even impossible. When it comes to such data, machine learning algorithms are of great importance in making decisions.

The main goal of the project was to classify the party affiliation of U.S congressmen as Democrats and Republican based on the dataset on the UCI website under the title congressional voting records dataset. The dataset is collected from 435 U.S House of Representatives Congressmen. The data collected is based on 16 key votes. The details for the votes are simplified as questionnaire answers and represented as “yes”, “no” and “unknown” or not answered on the data set. In the dataset, the row of the data represents the Congressmen, the first column is the label of the class (Democrat/ Republican) and the rest of the column is voting data (“Yes”, “No” and “Unknown”).

The purpose of the project was to predict the class of new data inputs based on the given data set in the future. The given dataset was trained with machine learning algorithms, so that the new observation can be predicted based on the previous knowledge of the trained data.

The whole dataset needed to be changed into nominal data for analytical purposes. Then the data was pre-processed to have the mean value of zero and standard deviation of one column wise. After that dimension reduction was done by removing some less informative features. This can be done by feature selection and feature extraction algorithms. Then the data was trained with machine leaning algorithms. There were different kinds of algorithms to choose based on the data, and the best algorithm gave the least classification error and that was selected.

Machine learning algorithms have been used in different areas of applications. The main application areas are analysis on large databases and on domains where human might not well establish hypothesis. In this project, classification for the party affiliation was analysed for 16 dimensional dataset for 435 observations using KNN and Naïve Bayes algorithm. The result was evaluated with the testing set and 95 % accuracy level was achieved.

Keywords	class, observation, KNN, attribute, classification, training set

Contents

1 Introduction

Machine learning algorithms play an important role in the analysis of huge databases which are cumbersome to human analysis. Massive amounts of data are collected and processed every day. In this process there might be a need for extracting a piece of information in a short period of time from junky data. Machine learning is a discipline that deals with such problems by designing algorithms in such complex and huge data. It gives efficiency to human learning, and gives structure and new findings which are unknown to humans. [1, 2-3]

Classification is one of the machine learning methods for identifying a new observation based on data training. It is a supervised learning, which means it gives prediction to new data from known input and known output data. [2, 5-6]

This project deals with classification of new observation, given the input which is the data and output which is the class. The dataset for this project is taken from the UCI website, which is a machine learning repository under the title Congressional Voting Record. The data is collected based on the voting of 16 key votes of US congressmen. The congressmen are subdivided into two political party classes, republican and democrats. The task of the project is to predict the class of new observation given the 16 key voting. Machine learning algorithms and techniques are used to tackle the problem.

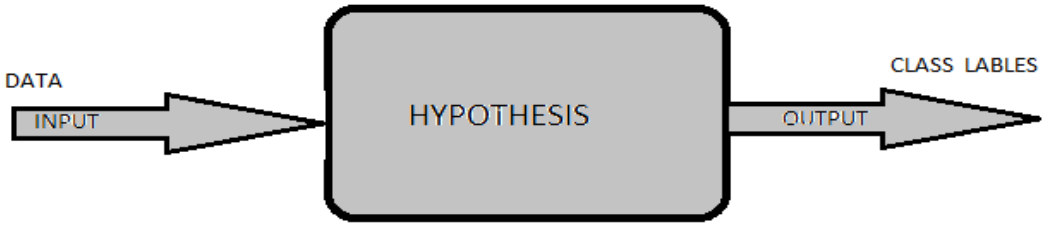

Figure 1. Input and output representation

Figure 1 shows the input-output model in a machine learning context. The input is data and the output is the class label, and the data undergoes some hypothesis to give the output.

2 Overview of Machine Learning

2.1 Input and Output Vectors

In a mathematical context a vector is a matrix with one row or one column. It is used in cases where a single variable could not explain the given set. In machine learning an input vector has different names, a feature vector, pattern vector, sample, example or instance. The components of an input vector are named as feature vectors, attributes, input variables and components. These names can be used alternatively or depending on the context. Components are divided into three main categories: real valued numbers where the components are real numbers, discrete value numbers where the components are distinct and categorical variables where the components are grouped or categorised in some pattern. The category can be nominal or ordinal. As an exception Boolean value can either be real value numbers as 1 and 0 or categorical variables as true and false. [1]

In order to get an output, the input has to go through some processes defined by a function. In this context there are also different names for the function and the output vector. Names depend on the output value. When the output is a real number, then the function is named as a function estimator and the corresponding output is called an output value or estimator. When the output is a categorical value, the function can alternatively be called a classifier, recognizer or categoriser and the corresponding output vector is called a label, class, category or decision. [1, 8-9; 3]

2.2 Learning Association

An association rule deals with the dependence of one variable on another variable. The basic principle of this rule is the conditional probability. In conditional probability, suppose we have variable X and Y, the conditional probability is defined as P (X|Y), read as probability of X given Y, where X and Y are defined in real number R. When the given probability is given as P (X|Y, D), then X and Y are defined in D dimension. [2, 3-4]

For instance, a customer buying a full suit is likely to buy a tie. The probability of buying a tie is conditional with buying a suit. The customer who buys a suit can be further identified by gender, age, career or marital status. This can be taken as a dimension of this category.

2.2.1 Supervised Learning

Supervised learning is a machine learning type where both input data and response to the data are known prior to any prediction. The task in the supervised learning is to find a predictor function or model that gives the best mapping for the new observation based on the training set. [2, 21-22] Figure 2 elaborates this learning diagrammatically. Suppose we have the following notations,

x_i – Input vector or feature
y_i – Output vector or target variable
(x_i, y_i) – Training examples
$\{(x_i, y_i)\}$ – Training sets
h - Hypothesis function or predictor model
where i = 1, 2, ---------- m – m is number of samples,

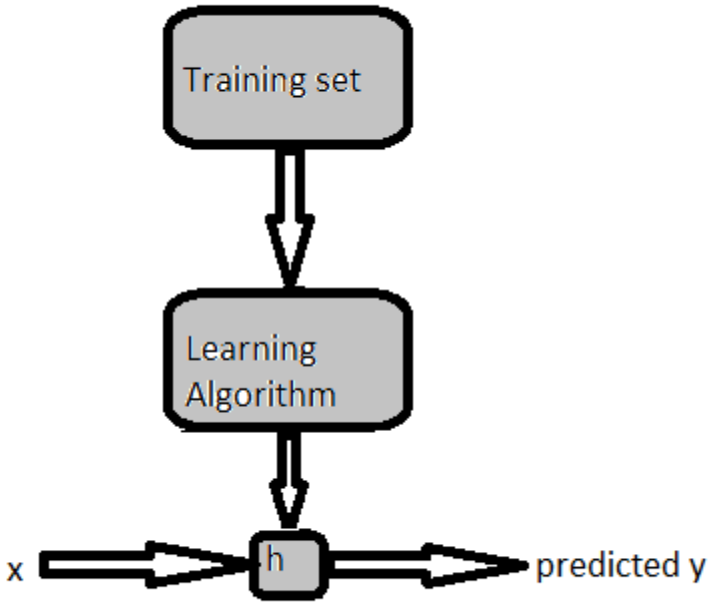

Figure 2. Supervised learning. Given the training set the learning algorithm finds the best hypothesis function h. Modified from Ng (2013) [4]

The best hypothesis is the one with good predicted values of y, especially for large numbers of training sets.

Notice that both x and y are real numbers and can be expressed in two dimensional space x and y. Take an example of a dataset of a living area and its corresponding selling price in the Helsinki area, randomly taken from an advertisement site for this example.

Table 1. Living area vs. price. Data gathered from Nettiasunto.com. [5]

Living area(m^2)	**Price(in thousands of euros)**
15	100
18.5	114,5
23	149
42	129
77.5	175
84,5	215

Table 1 shows the direct relationship between the living area and the price. The price is dependent on the area so the living area is the dependent variable whereas the price is the independent variable. The plot for this in the x-y axis looks like in figure 3.

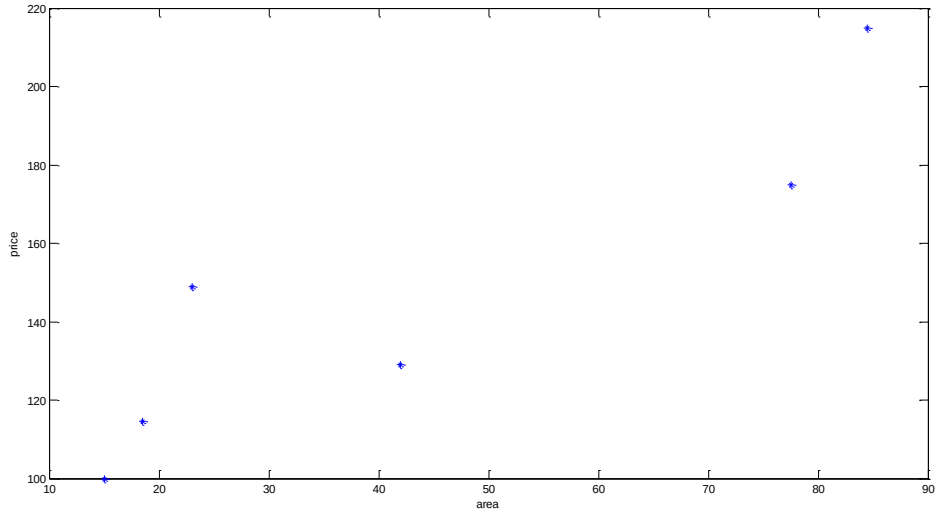

Figure 3. The area vs price plot for an apartment located in the Helsinki area, data gathered from Nettiasounto.com. [5]

If someone wants to know an approximate price of an apartment of a specific area, say 80 square meters, then there must be a predictor function which can predict well based on the dataset. Here comes the need for the training set. Based on the training set, a predictor function will be made. From this specific example, a graph like a linear function might be a good prediction or if it fits most of the training sets. A quadratic function of degree n can also fit well on the given data. For such a small dataset, choosing a model might be a tricky task because of over-fitting. Over-fitting is discussed in section 2.3.1.

In the example the price prediction is made based on the living area, but in reality the cost of a living place depends on many factors. The number of bed rooms, number of toilet, the availability of sauna, the view of the location, the proximity to public transport, the availability of a parking place and many more factors can affect the price. This is what we call the dimension of the data. When many factors affect the prediction, it cannot be represented on simple x-y axis, and simple models can not fit anymore. In this situation machine learning techniques are needed for prediction.

2.2.2 Unsupervised Learning

In the case of the unsupervised learning there are only input data in the learning processes. There are no output data given to map from the input data to output data like in the supervised data. The training data consists of only the input vector x. It does not have a target value to match. The aim of unsupervised learning is to find some regularity in the input data itself. In machine learning the processes of finding similarity in the training data is called clustering and in statistics the processes of finding the distribution of the input data is known as density estimation. [2]

2.2.3 Semi-supervised Learning

Semi-supervised learning is a method which lies between supervised and unsupervised learning methods. The goal of the learning is classification but the input data contains labelled and unlabelled data. In classification task there must be a labelled data. In semi-supervised learning, there exists a small amount of labelled data and a large amount of unlabelled data. The unlabelled data will be trained to learn the existing class so that classification is possible. [2]

2.2.4 Reinforcement Learning

Reinforcement learning is a learning method in which the output of the system is a sequence of actions. The learning algorithm finds the best action in order to get maximum reward. The best action is learned in the processes of trial and error. Each action in getting the maximum reward affects the current and the latter rewards in the processes. Playing a game can be a good example for reinforcement learning. A single move in a game is not important. It is a sequence of right moves that makes it good depending on the game policy. [2]

2.3 Model Selection

In model selection processes a data set is divided into three categories. Training set, testing set and validation set. The proportion of dividing the data is not distinct but the training set takes the lion share or the data and the smaller percentage of the data goes to testing and validation set.

2.3.1 Training Set

A model is selected for a specific problem from possible hypotheses. The best model generates the right output when it is used in the real world. A well-trained training set with the right model selection can easily predict the right output for a new observation. The goodness of the model to predict a new observation is called generalization. Generalization can be compromised with selecting a hypothesis that is less complex than the function or training set. In this case the hypothesis chosen does not even give the right response for the dataset. This is called under-fitting where we have a simple hypothesis for complex data. On the other hand selecting a complex hypothesis for a small data set can also compromise the data. In this case the selected hypothesis works perfectly for the given data set but the problem comes when it is tested for new instances. This is called over-fitting. [6]

2.3.2 Validation Set

A validation set is a part of the training set. The training set is further divided into training set and validating set in a certain way. The validation set is used in the model selection process. A cross validation method is one of the most common validation techniques. In figure 4 we can see the model selection section. In the case of cross validation with k folds, in this case k is 4 as we divide the whole data set into 4 sections. The whole data set is then divided in four equal folds. For a dataset with 80 instances and k = 4, then the dataset is divided into 20 k-folds. For the sake of simplicity k = 4 in figure 4.

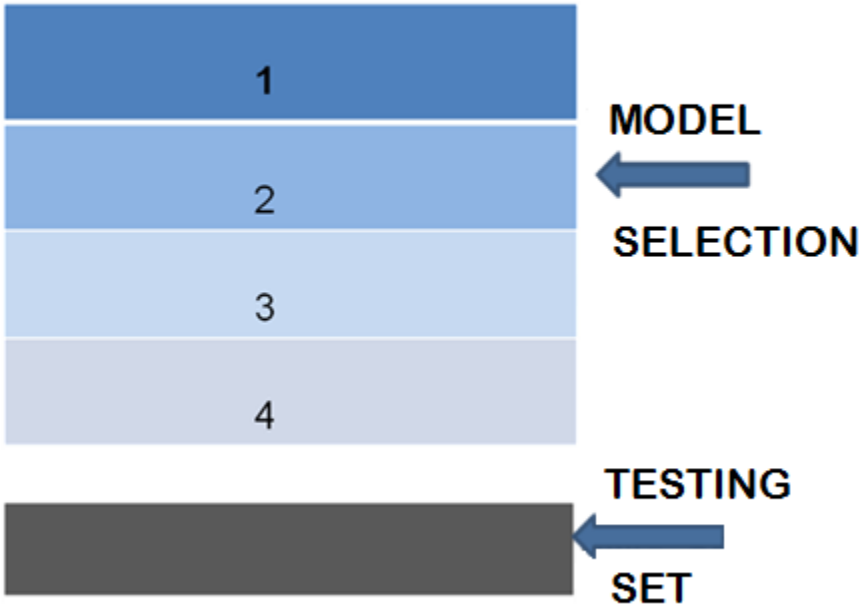

Figure 4. Model of model selection and testing set. Modified from Bishop (2006) [6]

In cross validation, referring figure 3, the step is as follows,

- 1 used as a validation set and 2, 3, 4 used as training set
- 2 used as a validation set and 1, 3,4 used as a training set
- 3 used as a validating set and 1, 2, 4 used as a training set
- 4 used as a validating set and 1, 2, 3 used as a training set

Then the validation error from each step is taken and the mean will be used as a final validation error. For datasets with large training set and validation set, the model with the least validation error will be selected as the best model for the given dataset. For a small dataset with a small training set and validation set, there might be a case of over-fitting problem. Minimal validation error might lead to the conclusion of best model selection. The problem comes when it comes to testing set, where it is tested with the real world data.

2.3.3 Testing Set

A testing set is a set of data which is not used either in the training or validation phase. This part of the data is not touched in the model selection processes. It is used as a simulation of the real world data. Usually in machine learning problems, dataset is divided in to 80 to 20 percentages. Where 80% is for the model selection processes and 20% is for testing proposes. The testing phase is needed to test the entire learning process, including the training phase and the model selection. Figure 3 shows the testing set separated from the model selection part to show that it is not used in the model selection process. [6]

3 Dimension Reduction Techniques

Numbers of input dimensions and data samples are the main factors that contribute to data complexity. The complexity of the data determines time and memory space resources for computation. For the sake of time and memory space resources reduction there is a need of dimension reduction in such a complex dataset. In dimension reduction, data is represented with relatively fewer dimensions without loss of information. In addition to saving memory space and time resources during computation, dimension reduction contributes to a better visualization and increases comprehensibility of the data. [2,109-110; 7]

There are two basic methods for dimension reduction. Feature selection which selects a subset of the original data and a feature extraction method which finds a new dataset with a smaller dimension that represents the original data.

3.1 Feature Selection

Feature selection is also known as subset selection since the method selects some but relevant subsets of the original data without losing information. Mathematically speaking, given feature set $X = \{x_i \mid i = 1, 2, - - - - - - - - - - - - N\}$ the feature selection finds subset $Y = \{x_{i1} x_{i2} - - - - - x_{iM}\}$ with M < N, that optimizes an objective function J(Y), denoting $\{x_{i1} x_{i2} - - - - - x_{iM}\} = argmax\ [J\{x_i \mid i = 1, - - - - N\}]$. Figure 2 shows this diagrammatically, and notice that $M < N$.

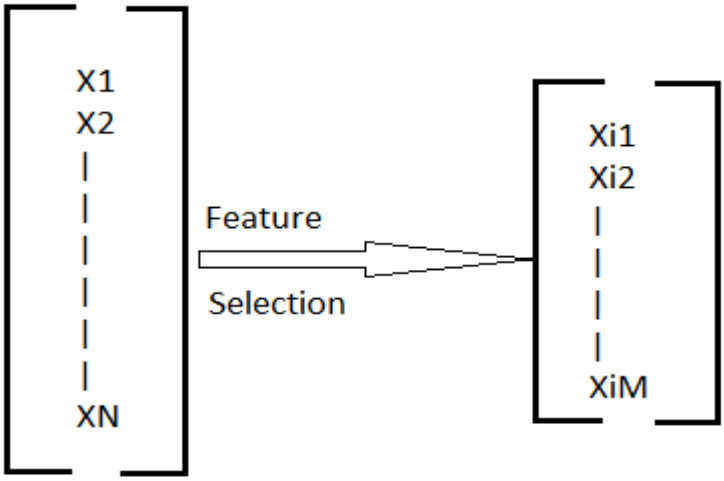

Figure 5. A feature selection taking the subset of the feature set to achieve dimension reduction. Adapted from Ricardo (2014) [8]

In feature selection there has to be a search strategy to select a candidate subsets and an objective function to evaluate these candidate. The objective function is the measures of the search strategy. There are two broad groups of objective function based on the evaluation method. They are filters and wrappers. In the case of filters, the objective function evaluates subsets by their information content. For instance, in the case of classification an interclass distance is used to classify the classes before the use of a learning algorithm. In the case of wrappers, the objective function is a pattern classifier and evaluation is based on predictive accuracy by cross validation. A learning algorithm is used in the subset selection process of wrappers method.

There are 2^N possible combinations of subsets for a given data set. For a larger dimensional dataset it is almost impossible to test every subset. So there is a need for a strategy search. One of the most used strategy search method is sequential feature selection. There are two kinds of sequential feature selection methods, sequential forward selection (SFS) and sequential backward selection (SBS). Sometime there might be a need to use a combination of both in a specific feature selection problem. [2]

3.1.1 Sequential Forward Selection (SFS)

A sequential forward selection method starts with empty subset attributes and add them one attribute at a time and evaluate them right away. Depending upon the evaluation the added attribute is kept or discarded. The adding process continues until no

attribute produces an improvement to the current subset. However this method suffers a nesting effect. This means it is unable to discard attributes that are added in some stage of the processes in a later stage. It works best for datasets having small number of attributes. The algorithm flow for SFS is shown below. [8]

1. Start with empty set $Y_0 = \{\phi\}$
2. Select the next best attribute$x^+ = argmax\ [J(Y_k + x)]$, where $x \neq Y_k$
3. If $J(Y_k + x^+) > J(Y_k)$, then
   Update$Y_{k+1} = Y_k + x^+$, $k = k + 1$
4. Go to step 2
5. Otherwise End

3.1.2 Sequential backward Selection (SBS)

The sequential backward selection method starts with the full subset and discards one attribute at a time. Evaluation is done at each step and discarding an attribute continues until the relevant attribute subsets are left. Each time if an attribute that is not relevant is discarded, then the value of the objective function also increases. The algorithm flow for SBS is shown below. [7]

1. Start with all attributes $Y_0 = x$
2. Removes the worst attribute $x_- = argmax\ [J(Y_k - x)]$ where $x \neq Y_k$
3. If$J(Y_k - x_-) > J(Y_k)$
   Update$Y_{k+1} = Y_k - x_-$, $k = k + 1$
4. Go to step 2
5. Otherwise End

This method suffers the lack of re-evaluating the relevance of the discarded attributes. It works best for datasets with large number of attributes. [8]

There are different kinds of sequential selection methods that combine the two methods, SFS and SBS, in order to overcome the drawbacks such as bidirectional search, sequential floating selection, Plus-L Minus-R selection. However the basics for this search lie in these two selection methods. [8]

3.2 Feature Extraction

Feature extraction is another method of dimension reduction techniques. In this method, instead of selecting the subsets from the existing dataset, it transforms the existing dataset by projecting to a new feature space. The new feature space is of a relatively smaller dimension with relevant information only.

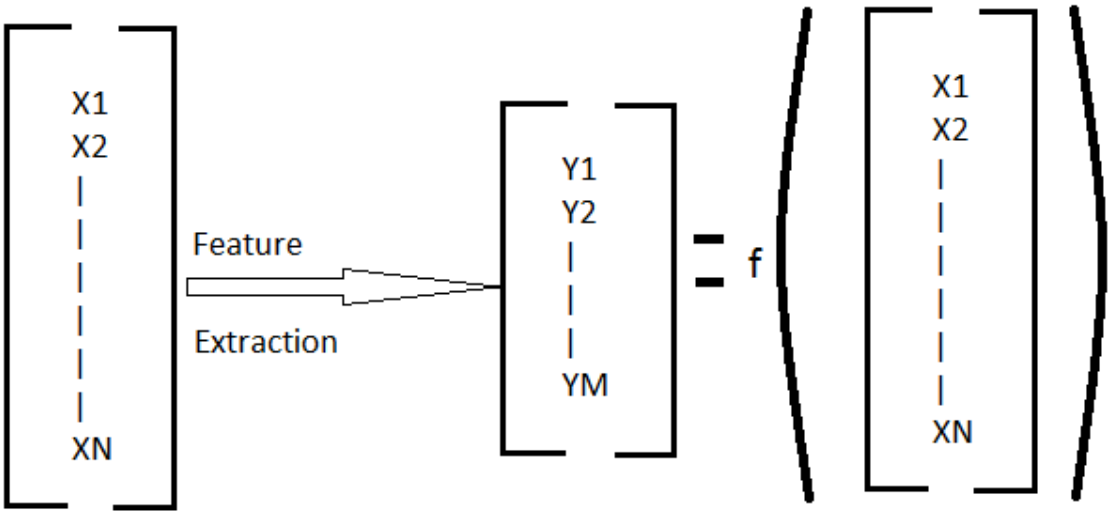

Figure 6. Feature extraction model. Adapted from Ricardo (2014) [8]

Figure 6 shows the dimension reduction method where X1, X2, --------, XN are transformed to new feature space Y1, Y2, --------, YM.

Principal component analysis and singular value decomposition are examples of feature extraction.

Principal component analysis (PCA)

PCA is a feature transformation method in which a new feature is created with lower dimension than those of the original feature sets. The new feature space selects features with more descriptive power and drops out the others. This method generates new variable sets which are a linear combination of the original variables called principal components. All components are orthogonal to each other in order to avoid redundant information and achieve a lower dimension as a result. [9]

In reducing n-dimension to k-dimension in PCA finds k vectors $u^1, u^2, - - - u^k$ on which to project the data in a manner of getting a minimum projection error. For instance, figure 7 shows dimension reduction of 2-dimensional spaces to 1-dimensional space.

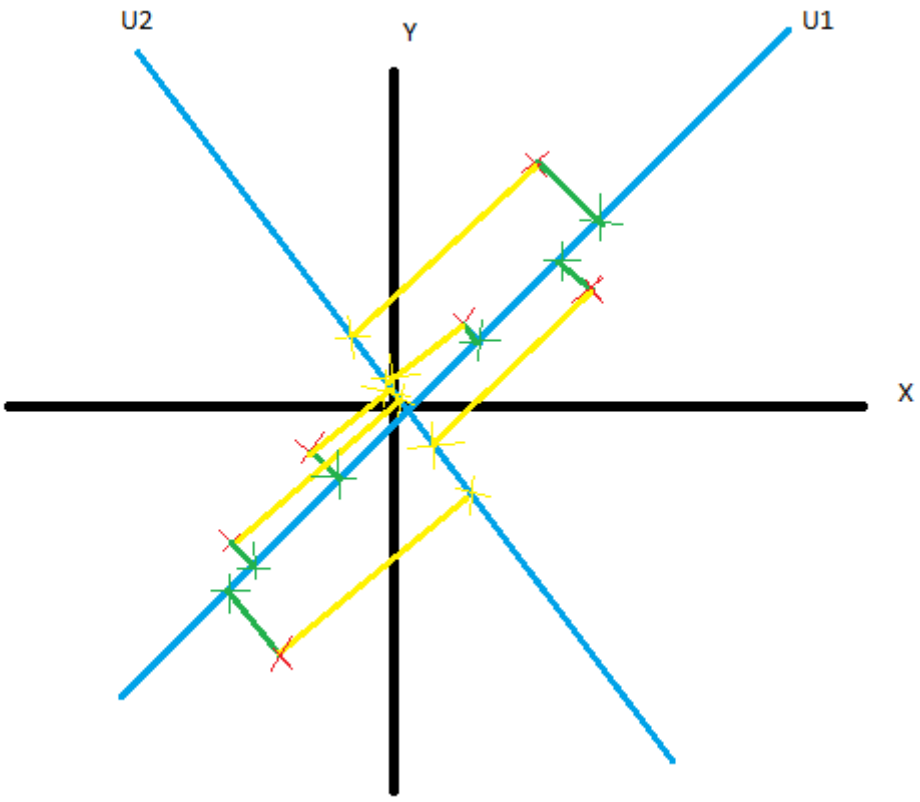

Figure 7. PCA representations when a 2-dimensional space is projected to a 1-dimensional space. Adapted from Ng (2013) [9]

The original data is indicated with red stars in the X-Y axis and U1 and U2 are the possible single dimensions to project the data. In the case of U1 it can be seen the projection distance which are the green lines are too small compared to the projection to U2 which are indicated with the yellow lines in figure 7. The aim of the PCA is to find the minimum projection error. So U1 is the best representation for the PCA. It can be noticed from figure 7 that the projected lines are perpendicular to the feature spaces U1 and U2 since the smallest distance to U1 and U2 are the orthogonal distances.

4 Parametric Method

In statistics, samples are taken from the data to make generalizations about the data. The common parameters of statistics like mean, variance, standard deviation and so on got from the samples are basically used to simulate the whole data. In the parametric approach of problem solving, it is assumed samples are taken from a distribution to obey some known model. An assumed parametric method determines the accuracy of the estimation. [10]

4.1 Probability Density Function (PDF)

In parametric density estimation, probability density functions are used for estimation. In the implementation of parametric method of density estimation, the basic task is to choose the probability density function. In the processes on selection, an assumption over some domain is taken. There are some commonly used PDFs like Gaussian, passion, uniform and so on. Some are continues over some range of values and others are discrete. After selecting the PDF the next task is to learn the parametric model based on the training data. [10]

4.2 Parametric Learning Approaches

There are three major categories in learning the parametric approaches with the training data. The maximum likelihood method that chooses parametric value to maximize the probability of the data, the Bayesian approach which maintains the probability distribution over all possible parameter values by balancing a prior distribution with the evidence of the data, and the maximum a-posteriori probability which compromise between the other two.[10]

4.2.1 Maximum Likelihood

The basic idea of the maximum likelihood method is to choose a parametric value that maximizes the probability of the observation. To elaborate this point, assume an independent observation $\{x^t\}$ drawn from some known probability density function $p(x|\emptyset)$ defined up to parameter $\emptyset$, for a dataset $X=(x^1, x^2 \ldots\ldots\ldots\ldots x^t)^N$. Based on

this assumption $x^t \sim p(x|\emptyset)$. The parameter needed is $\emptyset$, that makes the observation x^t, from the density function P $(X|\emptyset)$ as likely as possible. Since x^t are independent, the likelihood parameter $\emptyset$ given x written as $\ell(\emptyset|x)$ is the product of the individual likelihoods. Thus the maximum likelihood is given as

$$\ell\ (\emptyset|x) = p\ (X|\emptyset) = \prod_{n=1}^{N} p(x^t|\emptyset) \quad ---------- \textit{Equation } 4.1$$

Taking the logarithm for the whole equation changes the equation as follows without changing each value. $\mathcal{L}\ (\emptyset|x)$ is the likelihood after taking the logarithm equation.

$$\mathcal{L}\ (\emptyset|x) = p\ (X|\emptyset) = \sum_{n=1}^{N} \mathrm{P}\ (x^t|\emptyset) \quad ---------- \textit{Equation 4.2}$$

After this a PDF is selected for the particular problem. For instance, take the Bernoulli distribution with two outcomes which are independent.

$$P(x) = \begin{cases} p, when\ \ x = 1 \\ 1 - p, when\ x = 0 \end{cases} \quad ---------- \textit{Equation 4.3}$$

Where P is theprobability, then

$P(X) = P^{x}(1-\ P)^{\ 1-x}$ where $\ x\ \in \{0, 1\}$

The expected value which is the mean and the variance are calculated as follows:

$$E[X] = \sum_x xp(x) = p \quad ---------- \textit{Equation 4.4}$$

$$Var\ [X] = \sum_x (x - E[x])^2\ p(x) = p(1-p) \quad ---------- \textit{Equation 4.5}$$

Here the only given parameter is$p$ and the needed parameter is the estimator denoted as$\hat{p}$, the solve *Equation 4.4* for $\mathcal{L}\ (p|x)$,

$$\mathcal{L}\ (p|x) = \sum_t x^t \log p(N - \sum_t x^t) \log\ (1-p)$$

$\hat{p}$ that maximize the likelihood function can be found by taking the derivative of $d\mathcal{L}\ |dp\ = 0$, then the estimation $\hat{p}$ is,

$$\hat{p}\ = \frac{\sum_t x^t}{N} \quad ---------- \textit{Equation 4.6}$$

The estimate $\hat{p}$ is the number of occurrence of the event divided by the number of experiment N, with the assumption made above that X is Bernoulli distribution with p and *Equation 4.3*, the maximum likelihood estimator of the mean is the sample average. [2, 62-66]

4.2.2 Bayes Estimator

The Bayes estimator is used when there is prior information given about the parameter. The exact value of the parameter is not known but the range for the parameter is known. This is called prior information of the parameter. For a random variable θ which is the parameter, then $p(\theta)$ is the prior density. The prior density $p(\theta)$ is the likely value of θ may take before the sample observation. Combining the prior density with the likelihood density $p(X|\theta)$, using Bayes rule gives the posterior density of θ. Bayes rule states that,

$$posterior\ density = \frac{prior\ density * maximum\ likelihood}{evidence}$$

The posterior density $p\ (\theta|x)$, which is the value of $p(\theta)$ after the sample observation is given as follows:

$$p\ (\theta|x) = \frac{\mathrm{p}(X|\theta)\times \mathrm{p}(\theta)}{p(X)}$$

To find the Bayes estimator θ_{BAYES}, find the maximum argument for $p\ (\theta|x)$. The Bayes estimator then can be defined with the expected value of the posterior density.

$\theta_{BAYES} = \int \theta\ p(\theta|x) d\theta$. [2, 66-69]

5 A Non-Parametric Method

A non-parametric method is also called instance based or memory-based learning. The method assumes that similar inputs give similar outputs. The algorithm is dedicated in finding similar past instances from the training set based on distance measure and interpolating them in finding the right output. The fundamental difference with parametric method and non-parametric method is that no prior information is used in the non-parametric method. Density estimation is one of the most used methods in machine learning and statistical problems. From the given sample of observation density estimates describe the data in the best possible way as to some predefined criterions. In the next sections, some of the well-known density estimates are dealt. [11]

5.1 Histogram Estimation

Histograms are one of the simplest, widely used representations of densities. It is flexible and can model complex properties. In histogram density estimation, inputs are spaced into equal intervals called bins. Assuming a sample$X= \{x^t\}_{t=1}^{N}$ and given an origin x_0 and bin width h , the bin interval is given as $[x_0 + mh, x_0 + (m+1)h]$ for positive and negative value of m, and the density estimate$\hat{p}$is given as,

$$\hat{p}\ (x) = \frac{\#\{x^t \text{ in the same bin as } x\}}{Nh} \quad \text{---------- Equation 5.1}$$

While constructing a histogram the origin and the bin width are defined. The chosen origin affects the nearby boundaries of the bin but the estimate is mainly affected by the width of the bin. A naive estimator is introduced where x is always at the center of the bin and the equation is stated as,

$$\hat{p}\ (x) = \frac{\#\left\{x-h/2 < x^t \leq x+h/2\right\}}{Nh} \quad \text{---------- Equation 5.2}$$ [2, 165-167]

5.2 Parzen or Kernel Estimation

A Kernel estimator is a smother and contentious version of the histogram estimator. A smooth weight function called kernel function is used to get a smooth estimate. Given an independent sample$X= \{x^t\}_{t=1}^{N}$and considering $p_x$by a histogram formed from with bins that are $\Delta x = 2h$ wide as figure 8 shows. k is the number of samples in the bin and the mid-point $\hat{x}$. [2,166-168]

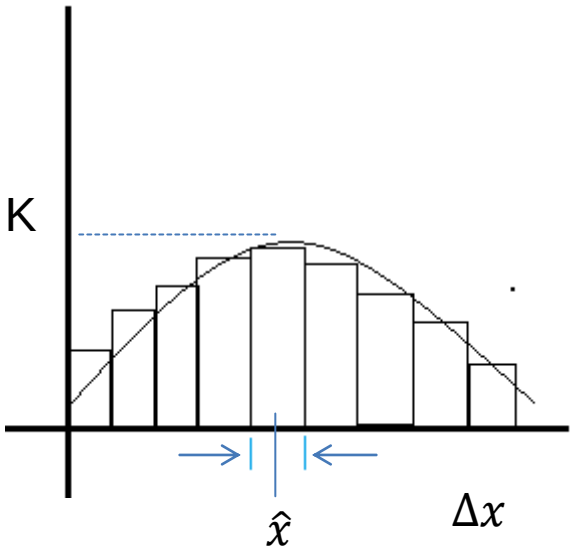

Figure 8. Approximation of density functions by histogram. Modified from Charles (1989) [12]

The probability that a sample is in that particular bin is approximated by

$$p[|\hat{x} - x| \leq h] = p_x(\hat{x})2h \quad \text{---------- Equation 5.3}$$

For a large number of samples *Equation5.3* is approximated by relative frequency k/N, and the density is approximated at $\hat{x}$ as

$$p_x(\hat{x}) = \frac{k}{2hN} \quad \text{---------- Equation 5.4}$$

Considering the function in *Equation5.4* the kernel function is plotted in figure 9

$$k(u) = \begin{cases} \frac{k}{2h} & , k \leq h \\ 0, otherwise \end{cases} \quad \text{---------- Equation 5.5}$$

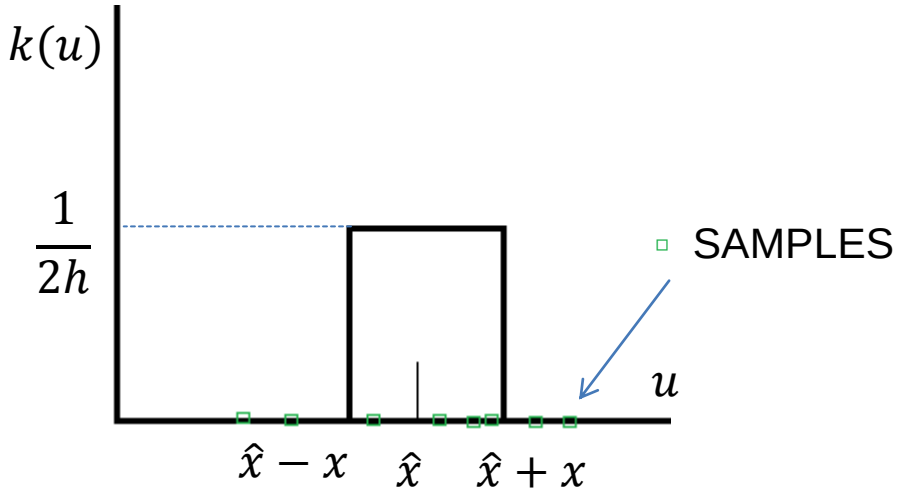

Figure 9. Rectangular kernel representation in one dimension. Modified from Charles (1989) [12]

One of the most popular kernel functions is the Gaussian kernel function,

$$k(u) = \frac{1}{\sqrt{2\pi}h} e^{\left[-\frac{u^2}{2h^2}\right]} \quad \text{---------- Equation 5.6}$$

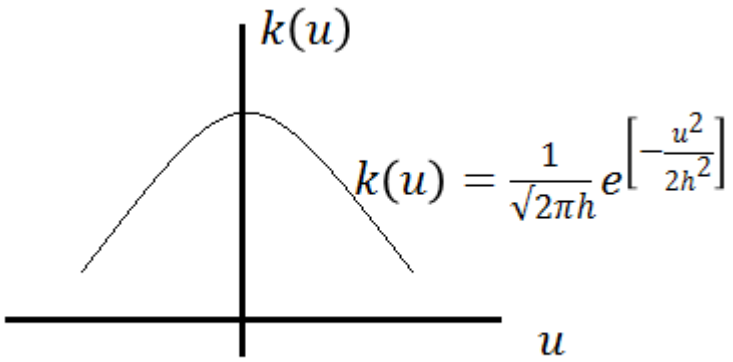

Figure 10. Kernel estimation for Gaussian distribution in one dimension. Modified from Charles (1989) [12]

Referring to figure 10 the kernel estimation for Gaussian distribution, the kernel estimator is then defined as,

$$\hat{p}\ (x) = \frac{1}{Nh}\sum_{t=1}^{N} K\left(\frac{x-x^t}{h}\right) \quad \textit{Equation 5.6}$$

The factor $K\left(\frac{x-x^t}{h}\right)$contributes to the smoothness and the window width h determines the width. [12]

5.3 K-Nearest Neighbour Estimation

The k-nearest estimate is an extension of the Parzen density estimate. In the k-nearest estimation, the number of samples will be fixed and the region around $\hat{x}$ contains k samples. The degree of smoothing is dependent on k. The k-nearest neighbour density estimate is defined as,

$$\hat{p}(x) = \frac{k}{2Nd_k(x)} \quad \textit{Equation 5.7}$$

Where $d_k(x)$ is defined as the distance between two points.

In the case k-nearest neighbour density estimation, instead of fixing h and checking how many samples fall in the bin, it fixes k which is the number of observation to fall in the bin and it computes the bin size. When the bin size is small the density is high, whereas when the bins are large the density gets low. The KNN estimation is not continuous, since its derivative has discontinuity. Notice that the KNN is not PDF since it integrates to ∅. [12,125-131; 2,168-170]

6 Performance Evaluation Methods

6.1 Mean square error (MSE)

A mean square error is one of the criteria to fit observation to a model. Suppose there is an estimate$\hat{\theta}$ for an unknown parameter θ for a random sample X =$\{x_1, x_2, \ldots\ldots\ldots x_N\}$, then the deviation of the estimate to the true value is $|\hat{\theta} - \theta|$. This measures the quality of the estimator. [12] For computational purposes, taking the square $(\hat{\theta} - \theta)^2$ doesn't affect the quality measure. Function $E(\hat{\theta} - \theta)^2$is called the expected value of the mean square or risk function of an estimate.$(\hat{\theta} - \theta)^2$ is called quadratic loss function. Since X is a random sample, θ is an average value of the sample.

Let us see the analytical advantage of taking the mean square error $E(\hat{\theta} - \theta)^2$ instead of the absolute error $|\hat{\theta} - \theta|$ even though the absolute errors can also tell the quality of the estimation.

Mean square error ($MSE_{\hat{\theta}}$) = $E(\hat{\theta} - \theta)^2$= $E\ (\hat{\theta}^2)$ +$E\ (\theta^2)$ - $2\theta E(\hat{\theta})$

$= var(\hat{\theta}) + \left[E(\hat{\theta})\right]^2 + \theta^2 - 2\theta E(\hat{\theta}^2)$

$= var(\hat{\theta}) + \left[E(\hat{\theta}) - \theta\right]^2$

The term $E(\hat{\theta}) - \theta$ is called bias of $\hat{\theta}$ for the parameter θ. Bias is the measure of accuracy. When $E(\hat{\theta}) - \theta$= 0 then $E(\hat{\theta}) = \theta$ which means $\hat{\theta}$ is unbiased estimator of θ and it is an accurate estimate. A high biased error is sign of an under-fitting problem. This happens when a very simple model is used in a large dataset. This leads to a very large $E(\hat{\theta}) - \theta$. On the other hand the term $var(\hat{\theta})$ is a measure of the variability of the estimate that tells about the precision of the estimator. High variance is the problem of over-fitting where a higher order of model is used for small dataset. In this case the error in the training set is very small but the error in the testing set will very high. This is the cause of the high variance. Optimized variance and bias error contributes an optimized mean square error and can be generalized as a good learning algorithm. [13]

6.2 F-score

The F-score measures the performance of machine learning algorithms based on precision and recall. Accuracy is a measure of performance when there are no labels for different classes. On the other hand precision and recall are evaluation methods for the classifier's performance. In a classification problem, the evaluation of the predicted class and the actual class fall in four categories; true positive, true negative, false positive, false negative. The "positive" and "negative" refer to the classifier prediction whereas "true" and "false" refer to the observation. Based on these facts the "precision" and "recall" are "defined" as follows,

$$precision = \frac{true\ postive}{true\ postive + false\ postive}$$

$$recall\ = \frac{true\ postive}{true\ postive + false\ negative}$$

F-score combines precision and recall with the parameterβ,

$$F - score = \frac{(\beta^2 + 1) * precision * recall}{\beta^2 * + precision}$$

"Precision" is a function of "true positive" and of an observation misclassified as positive. "Recall" is a function of a correctly classified observation, "true positive" and its misclassified examples, "false negatives". The F-score is balanced when the factor β = 1. It factors the recall when β>1 and it favours precision when β< 1. The F-score gets its best result when it approaches 1 and the worst result when it approaches zero. [14]

6.3 Confusion Matrix

A confusion matrix gives one way to measure the quality of a classifier. The measure of the quality is based on a two-by-two confusion matrix recording the correctly and incorrectly classified observations for each class. The column of the matrix represents the classified observation whereas the row represents the actual class.

Table 2. Confusion matrix Modified from Hamilton(2012) [15]

ACTUAL CLASS	PREDICTIVED CLASS	
	TRUE POSITIVES	FALSE NEGATIVES
	FALSE POSITIVES	TRUE NEGATIVES

Table 2 is simplified to two by two matrix, confusion matrix. “Positive” and “negative” refer to the classifier prediction whereas “true” and “false” refer to the observations. Table 2 is interpreted as follows,

- “True positives”: number of correctly predicted classes for a positive instance
- “False positives”: number of incorrectly predicted classes for a positive instance
- “False negatives”: number of incorrectly predicted classes for a negative instance and,
- “True negatives”: number of correctly predicted classes for a negative instance.

Different performance criteria can be drawn from the confusion matrix, including precision and recall. Accuracy of the performance is determined as follows, [15]

$$Accuracy = \frac{True\ positive + True\ negative}{True\ positive + True\ negative + false\ positive + false\ negative}$$

6.4 ROC curve

The ROC curve, Receiver Operating Characteristics curve is a graphical way of examining the performance of classifiers. The ROC curve graph is a plot representation with false positive rate on the x axis and true positive rate on the on the y axis. The point (0, 1) corresponds to a perfect classifier. It classifies all the positives and negatives correctly, since the false positive rate is zero and everything else is the true positive rate. The point (1, 0) corresponds to incorrect classification for all cases. [15]

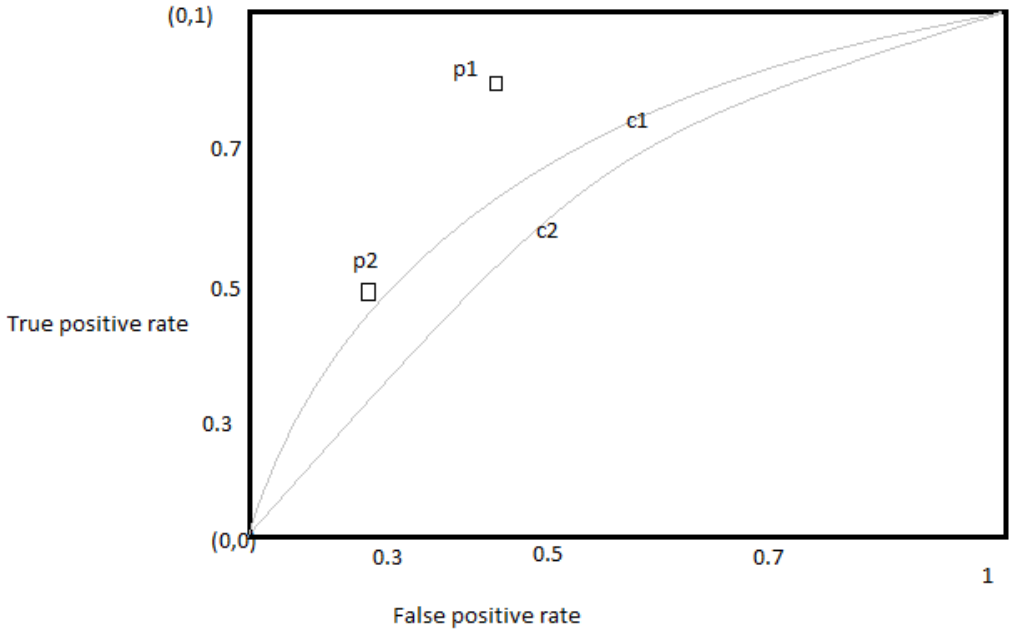

Figure 11. ROC curve with two class labels c1 and c2 and with two ROC points labelled as p1 and p2.

Figure 11 shows the model of the ROC curve. Besides the points the mentioned area under and above the curve also tells about the performance. For instance the area under the ROC curve is a measure for the accuracy.

7 Methods Used

7.1 Tools and Approach

This paper is based on the project done on predictive techniques for voting affiliation on American congressional voting based on machine learning algorithms. The voting record is collected from a machine learning repository site UCI. UCI is a machine learning repository for educational purposes with a variety of datasets. The dataset can be found under the title Congressional voting records on UCI database. The sample of the row data is shown in appendix 1. The data is collected from 435 U.S House of Representatives congressmen based on 16 key votes. For each 16 attributes there are 9 different types of votes. However 9 votes are simplified to 3 in the following manner.

- voted for, paired for and announced as yes
- voted against, paired against and announced against as no
- voted present, voted present to avoid conflict of interest and did not vote as unknown.

In appendix 1 votes are represented as “y”, “n”, “and “?”. The parties for each congressman are also represented as “republican” and “democrat” prior to each vote. In appendix 2 the information for the attributes are shown. Each vote is based on the opinions for each 16 questionnaires. The first attribute information in appendix 2 is the party name and it is called class in the machine learning language. The main task for this project is to predict the class of a new observation based on 16 attribute information in the future. In order to do that, the given data was trained with a machine learning algorithm. Based on the learned algorithm it predicts the class for a new observation. It is supervised learning since the input attributes and the output of the data which is the class are given prior to the prediction.

MATLAB was used as a tool for this project. MATLAB is a high level computing language and interactive environment for algorithm development. It has an interactive environment for numerical computation, visualization and programming. Its tools and built-in functions enable to explore multiple approaches faster than with traditional programming languages. [17]

7.2 Pre-processing

Pre-processing of the data has a significant impact on the generalization performance of the machine learning algorithm. Pre-processing of the data deals the following points.

- Instance selection and outliers detection which deals with the processes of unknown feature values.
- Missing feature values from the data, incomplete data is unavoidable but knowing the source of missed data helps how to treat the data.
- Data normalization which is a scale minimization method.
- Feature selection which is the processes of identifying a feature's relevance, irrelevance and redundancy.
- Feature construction or transformation which leads to transformation of a feature to a new feature set which is more concise and descriptive. [18]

The data taken from the UCI has to go through some pre-processing stage in order to use it in machine learning algorithms. The first pre-processing stage is to change the given data to nominal values. "y" is changed to 10, "n" changed to 5 and "?" changed to 0. Notice that the nominal representation is needed only for calculation purposes and the magnitude does not have an effect on the data. A sample of the nominal representation of the data set is shown in appendix 3. The missing attributes do not mean that are unknown instead they are neither "yes", nor "no". The class distribution is 45.2 % for democrats and 54.8 % for republicans.

Data normalization is needed in order to avoid an instance of a feature which has a larger magnitude of variance. If the variance is of considerably large magnitude, it might dominate the objective function. This makes the estimator unable to learn from other features correctly. For this purpose data is normalized to have a mean value at zero and standard deviation of one.

After normalizing the data, the next step is dimension reduction. Most datasets have huge dimensions. The machine learning process for huge data needs time and memory resources. To avoid time needed and memory space during the training phase dimensions are reduced without information loss. In this project two methods of dimension reduction are dealt, sequential backward selection and principal component analy-

sis. The former one is a feature selection method whereas the latter one is a feature extraction method. The dimensions of the dataset used in this project are435x17. The dataset is structured to 435 observations, 16 attributes and one dimension for the class label. The dimension reduction is needed for the attribute part. In MATLAB context reduction is done column-wise. The whole 435 observation is taken for this dataset. If the set of observations is too large a sample can be taken from the observations. The first column is the class label and the other 16 are the attributes.

7.3 Classification

Classification has two distinct meanings. One is establishing the existence of a class or a cluster in a data from a given set of observation. This method of classification is called clustering. This time the task is grouping similar objects into groups. Clustering is unsupervised learning method. The other one is the task of identifying the outcome class for a new observation based on the training set of data whose class categories are already known. The given dataset is divided into a training set, a validation set and a testing set. The training set is used to train the outcome to a given data. The validation set is used in the model selection process and the testing set is used to simulate the new observation. In this project the focus was on the latter one and the term classification is used to refer the supervised learning. Two classification algorithms were used: KNN (k-Nearest Neighbours) and Naive Bayes classifier. The former one is non-parametric method and the latter one a parametric method. Both are discussed in the following subsections.

7.3.1 KNN (k-Nearest Neighbours)

KNN is a non-parametric method of machine learning algorithm. There is no theoretical assumption of probability density function and requires no model to fit. It is a very lazy algorithm since there is no explicit training phase or it is very minimal and fast if there is a need. Classification is achieved by the majority vote of its neighbours. Distance is used as a mechanism for assessing the similarity among classes. Euclidean distance one of the most used distance measure in this case. [19]

Given data $D = (x_1, y_2), - - - - -, (x_n, y_n)$where $x_1, x_2, - - -, x_n$ are the observation in $d -$ dimensional space then $y_1, y_2, - - -, y_n$ the corresponding classes and$x_i \in \mathbb{R}^d$,

$y_i \in \{0,1\}$ assuming binary classification. The task of KNN algorithm is to find the corresponding value of y for a new observation x. For simplicity features are assumed in $\mathbb{R}^2$ and Euclidian distance in feature space is used. The Euclidian distance is defined as $d(x_i, x_j) = \|x_i - x_j\| = \sqrt{x_i^2 - x_j^2}$. In KNN, since there is no training phase the entire data is simply stored at this phase. At the testing phase a new observation $\hat{x}$ is selected. To predict the label of the class for the new observation $\hat{x}$, a training example x_i is taken based on its similarity to the new observation $\hat{x}$. Similarity in this contex is the minimum distance between $\hat{x}$ and x_i . Since x_i is the training example it has the class label y_i . Based on KNN algorithm the new observation $\hat{x}$ which has a minimum distance with the training example (x_i, y_i) is predicted to have the same class label y_i . In figure 8 KNN is explained for binary class 0 and 1in two dimensional space.

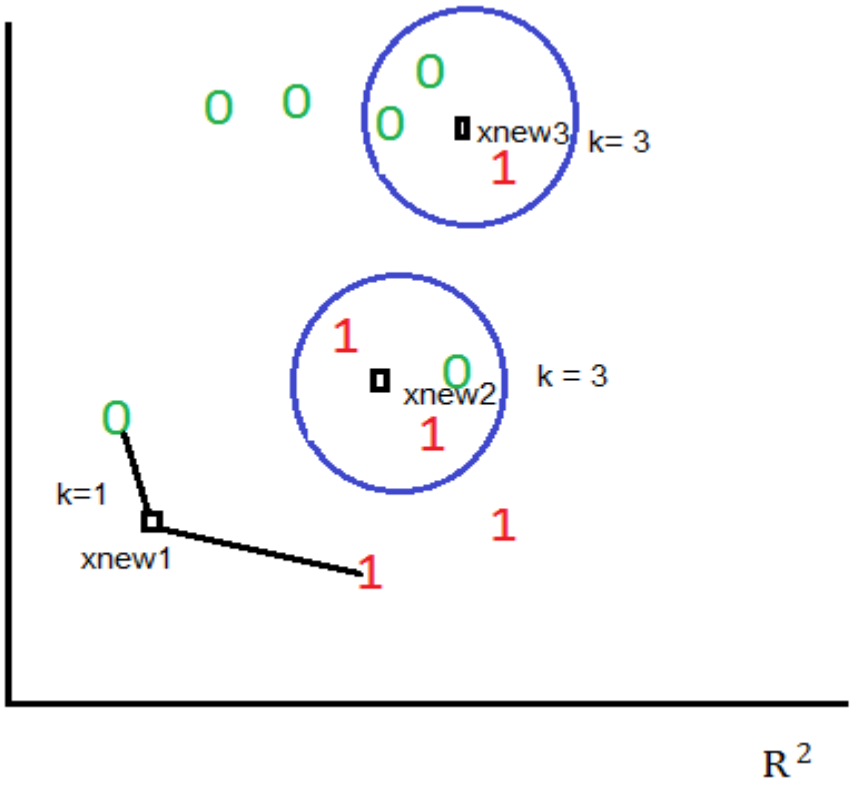

Figure 12. KNN model for three different new observation xnew1, xnew2, xnew3 with k = 1 and k= 3. Modified from [21]

Figure 12 explains KNN with three different observations xnew1, xnew2 and xnew3. In the case of xnew1 and k =1, there are two lines projected to measure the nearest distance to training examples. It can be seen that the new observation xnew1is closer to 0 than 1. So xnew1 is simply classified as 0. In case of xnew2 and xnew3, k=3: to calculate the nearest distance a circle is drawn centered at the new observations xnew2 and xnew3 and set to include three observations. As a result 1 and 0 are the class labels for new observations xnew2 and xnew3 respectively.

One issue in the implementation of KNN algorithm is how to choose the value of k. With a very small k, there is a risk of over-fitting whereas with a very large k, KNN predicts the majority vote and there is a risk of under-fitting. Therefore k is considered as a hyperparameter of KNN algorithm as a trade-off between over-fitting and under-fitting. To avoid this k is selected based on the least error in the validation set.

KNN assumes the nearby points should have the same label. If the data have few relevant features and lot of irrelevant features the KNN works poorly. Feature selection and feature extraction should be done before the implementation of the algorithm. Feature scaling is equally important, if features are in different measuring scales. Features in different measuring scales might lead to a wrong conclusion in the learning process.

7.3.2 Naive Bayes Classifier

In the case of the Naive Bayes classifier, the classification technique is based on the Bayes theorem with the assumption of independent features. Features are unrelated to each other and conditional with the class label. Thus the classifier assigns a new observation to the most probable class assuming features are conditionally independent given the class label.
Bayes theorem based on the probability states;

$$posterior = \frac{prior * maximum\ likelihood}{evidence}$$

The Naive Bayes classifier is also based on the assumption of PDF. Different models are chosen depending upon the nature of the dataset to fit the model. Given class labels ω_1 and ω_2 and observation $\hat{y}$, Bayes classifier can be stated as,

$$p(\omega_k|\hat{y}) = \frac{p(\omega_k)*p(\hat{y}|\omega_k)}{p(\hat{y})}\ ,\ \ k = 1{,}2 - - - - - - - - - - - - - - - - - - Equation\ 7.1$$

Since the observations are assumed to be independent

$$p(\hat{y}) = \ p(\hat{y}|\omega_1) * p(\omega_1) + \ \ p(\hat{y}|\omega_2) * p(\omega_2) - - - - - - - - - - - Equation\ 7.2$$

Combining $Equation\ 7.1$ and $Equation\ 7.2$ gives,

$$p(\omega_k|\hat{y}) = \frac{p(\omega_k)*p(\hat{y}|\omega_k)}{pp(\hat{y}|\omega_1)*p(\omega_1)+\ p(\hat{y}|\omega_2)*p(\omega_2)}\ ,k = 1{,}2 - - - - - - - - - - Equation\ 7.3$$

Then the decision of the class depends on the conditional probabilities

If $p(\omega_1|\hat{y}) > p(\omega_2|\hat{y})$ then class label ω_1 is selected other wise ω_2 is selected

One can use Naïve Bayes without Bayesian probability. In that case the ratio of maximum likelihood is used to make the decision. The likelihood ratio for class $\omega_1\ and\ \omega_2$ with the same observation $\hat{y}$ is as follows.

$$\mathcal{L}(\hat{y}) = \frac{p(\hat{y}|\omega_1)}{p(\hat{y}|\omega_2)} > \frac{p(\omega_2)}{p(\omega_1)}$$

If $\frac{p(\hat{y}|\omega_1)}{p(\hat{y}|\omega_2)} > \frac{p(\omega_2)}{p(\omega_1)}$ then class ω_1 is selected, otherwise class ω_2 is selected.

The classification with Bayesian probability or maximum likelihood is nearly identical. The fundamental difference is, in case of the maximum likelihood the parameter vector is fixed whereas in the case of Bayesian, the parameter vector is considered a random variable. The training data allows the conversion of the distribution in the parametric vector to posterior probability density.

8. Result and Analysis

This section deals with the results gained in the project. After pre-processing the row dataset, dimension reduction was done. Two methods of dimension reduction were implemented, sequential backward selection and PCA. In the case of sequential selection method the 16 columns were reduced to 6 columns. The sequential backward selection method takes the subset of the original dataset with six columns of relevant information. The best selected features are columns labelled with [3 4 8 9 10 15]. This has been selected after running the sequential forward selection method several times. The columns selected happened to be found in most executions.

The other dimension reduction method used was the PCA. The PCA stores the most informative columns in the first few columns. The first few columns can tell most of the information in the dataset. While analysing the PCA function in MATLAB Eigen analysis of the covariance is used. In MATLAB eig function, order of information in the principal components is stored in the last columns when reading from left to right.

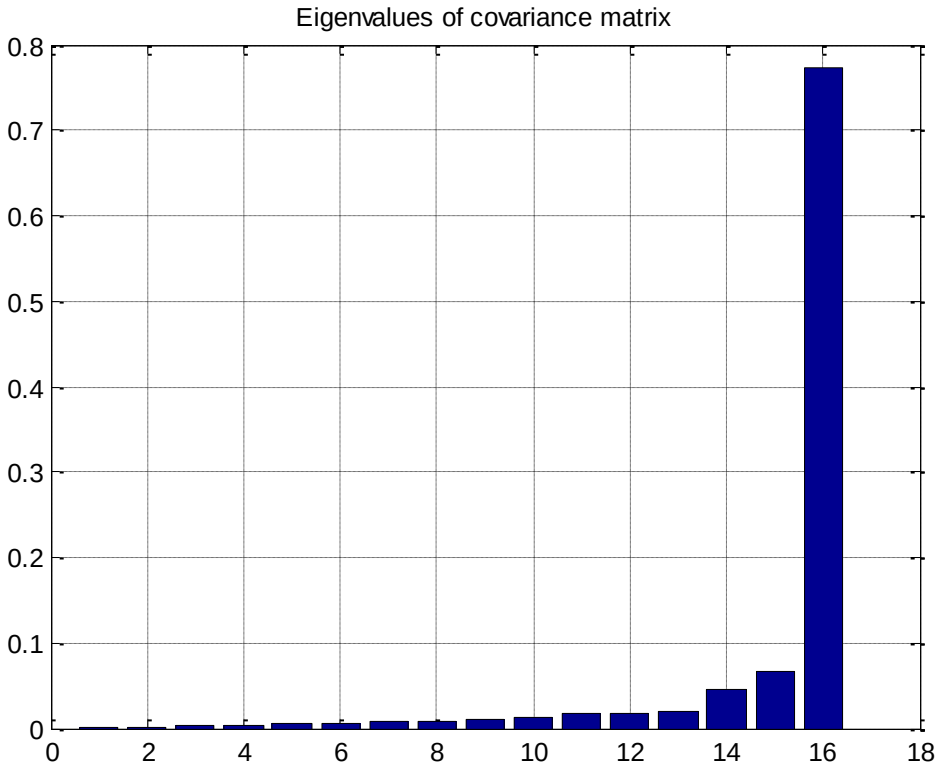

Figure 13. The Eigen values of the covariance matrix

Figure 13 shows the Eigen values for the columns. The Eigen values in the last columns are more dominant than the ones at beginning of the column. So the most informative principal components in the dataset in decreasing order are 16, 15, 14, --------, 2, 1. Column 16, 15, 14 and 13 are taken and the 16 feature spaces are minimized to 4 feature spaces.

The next phase was to use the learning algorithms with the reduced dataset. Two machine learning algorithms are used in order to compare the accuracy of the result, KNN and Naive Bayes classification.

The aim of the entire project was to classify a new observation based on the leaning algorithms. In order to achieve that, the data was divided in two parts, training data and testing data. The dataset has 435 observations, and 335 of the observations from the dataset were taken to train the data and the other 100 were taken to test the learning algorithm. The testing set taken simulates new observations. The testing sets should go through the pre-processing stage to have the same dimension with the training set. However it should not have to go through any training phase.

A confusion matrix was used as an evaluation method, and the results for each technique were revised in table 3. The table shows the percentage of correctly classified observation.

Table 3. Evaluation of each learning algorithm with the respective reduction technique

Learning algorithm and dimension reduction method used	Accuracy of the predicted class based on confusion matrix (%)
KNN and SBS	95
KNN and PCA	93
Naive Bayes classifier and SBS	88
Naive Bayes classifier and PCA	90

Referring to table 3, for the dataset used in the project, KNN with the SBS reduction technique gave the best result. Comparing the learning algorithm used, KNN gave a better result than Naive Bayes. Evaluating the dimension reduction technique from this table is impossible since the table evaluates both the dimension reduction and the

learning algorithm together. In machine learning achieving more than 80 % classification accuracy is considered as good prediction.

In order to compare the significance of the dimension reduction let us see the following table without the dimension reduction techniques.

Table 4. Accuracy of the predicted class without dimension reduction techniques

Learning Algorithm without dimension reduction	Accuracy of the predicted class based on the confusion matrix (%)
KNN	92
Naïve Bayes	89

Table 4 shows that KNN still gives a better prediction than the Naïve Bayes classifier. However, comparing table 3 and table 4, there is a decrease in the accuracy of KNN without the reduction method. In the case of Naïve Bayes classifier, the accuracy is almost the same with or without the reduction technique.

In the case of KNN irrelevant or redundant information may lead to a wrong conclusion. KNN is very sensitive and reduction is a must before using the algorithm. In this dataset the dimension is 16, and it can be seen that the difference in the accuracy is not significant. As the dimension increases the effect on the accuracy level increases.

Naive Bayes classifier is basically recommended for a very huge dataset. The significance of the dimension reduction might be high in huge datasets. Usually 16 feature spaces are considered as moderate feature space. The dimension reduction in the data set does not show the significance on the accuracy of the result.

Dimension reduction in the case of PCA is guaranteed if the original dataset is correlated. If features are uncorrelated in the original dataset PCA just groups the data with the variance and reduction cannot be achieved.

However the purpose of the dimension reduction is not entirely related to accuracy of the result at all. The need to reduce the dimension comes with the question of execution time and memory resources. In addition to this point, dimension reduction contributes to a better visualization and increases comprehensibility of the data.

9 Discussion

Data is collected for different reasons in many different areas. Experts then analyse the given data to draw some conclusions in different sectors such as business, economic, heath, and politics. All data collected might not be relevant and some analysis may lead to a wrong conclusion if it was not done properly.

In this project the data was collected from 435 U.S House of Representatives Congressmen based on 16 key votes. The purpose of analysing the data was to classify the party labels based on the 16 key votes for new observations. The results found in this project could give significant information in different sectors. One can predict the presidential voting from the outcome or draw some conclusion on the immigration system or the foreign policy. Experts on politics can draw many conclusions from the outcome of the project. The techniques used in this project can be used in any classification problems. Depending on the dataset the output may vary. Most of the steps and algorithms used in this project can also be used in other datasets.

The main challenge for the project was visualization of the whole dataset. The data had 16 dimensions. After the pre-processing stage the data was decreased to 4 and 6 in the two used dimension reduction methods. Finding out which attributes made an influence on the data during the dimension reduction method was a challenge. Frequency of each attribute, probability density and correlation of the individual question were studied to make some simplification on the challenge.

The objective of the task in this project was to be able to make prediction on voting affiliation of the American congressmen based on the machine learning algorithms. The task of classification is done based on two algorithms. Achieving a 95 % accuracy level is success for the project. For a huge dimensional data 80% of accuracy is usually considered as a successful classification task. For such a moderate size dataset 95 % accuracy is an acceptable result.

Machine learning is a field that has a room for an improvement. This particular project might give a better accuracy depending on the learning algorithm used and different approaches for the pre-processing stages. One can further take the project to a better accuracy level.

10 Conclusion

The goal of the project was to predict the party affiliation of the U.S House of Representatives Congressmen based on the 16 key votings using machine learning algorithm. The collected data has gone through pre-processing and training phase in order to make a prediction for new observation of data. A testing set is used to make a real word data simulation and to evaluate the models used.

To be able to make predictions of the class of the party based on the collected data has significance in many sectors. To understand and analyze the voting records, it needs some understanding of U.S Congress voting system and the legislative processes. However to be able to have a predictive result on hand before any circumstances is an advantage. The scope of this project is to make the prediction based on the previous knowledge from collected data and to implement a machine learning algorithm.

The analysis and the testing phase were done with a few observations. The project could be tested with more new observations and further evaluation could be done. Further study with different machine learning algorithms could be done to achieve a better accuracy. The overall result was a success with an excellent accuracy.

References

1. Nils Nilsson. Introduction to Machine Learning | PDF, EPUB, DOC Free Download EBook and Audiobook [Online].
   URL: http://getebook.org/?p=322170
   Accessed 20 February 2014

2. Ethem Alpaydm. Introduction to Machine Learning. Second edition. Massachusetts London, England: The MIT Press Cambridge; 2010.

3. Australian Bureau of Statistics. Numerical Data: What's The Difference between Discrete and Continuous? [Online].
   URL:http://www.abs.gov.au/websitedbs/CaSHome.nsf/Home/CaSQ+3B+NUMERICAL+DATA:+WHAT'S+THE+DIFFERENCE+BETWEEN+DISCRETE+AND+CONTINUOUS
   Accessed 20 February 2014

4. Ng Andrew. Machine Learning, Supervised learning [Online]. 2013
   URL: http://cs229.stanford.edu/notes/cs229-notes1.pdf
   Accessed 25 February 2014
5. Apartments for sale.
   URL: http://www.nettiasunto.com/en/advSearch.php?hty=apt&ctg=S
   Accessed 20 March 2014

6. Bishop Christopher. Pattern Recognition and Machine Learning (Information Science and Statistics). Secaucus, NJ, USA: Springer-Verlag New York, Inc.; 2006.

7. Feature Selection [Online]. MathWorks; 2014.
   URL: http://www.mathworks.se/help/stats/feature-selection.html
   Accessed 2 March 2014

8. Gutierrez Ricardo. Introduction to Pattern Analysis 1, LECTURE 11: Sequential Feature Selection [Online]. Texas A&M University;
   URL: http://www.facweb.iitkgp.ernet.in/~sudeshna/courses/ML06/featsel.pdf
   Accessed March 2014

9. Ng Andrew. Machine Learning, Principal components analysis [Online]. 2013.
URL: http://cs229.stanford.edu/notes/cs229-notes10.pdf
Accessed 25 February 2014

10. Parametric Density Estimation and Bayesian Learning [Online].
URL: http://isites.harvard.edu/fs/docs/icb.topic539621.files/lec16.pdf
Accessed 11 April 2014

11. Kontkanen Petri, Myllymäki Petri. MDL Histogram Density Estimation. Complex Systems Computation Group (CoSCo) [Online].
URL:http://machinelearning.wustl.edu/mlpapers/paper_files/AISTATS07_KontkanenM.pdf
Accessed 11 April 2014

12. Charles Therrien . Decision, estimation, and classification: an introduction to pattern recognition and related topics. Wiley; 1989.

13. Songfeng Zheng. Math 541: Statistical Theory II, Methods of Evaluating Estimators [Online].
URL:http://people.missouristate.edu/songfengzheng/Teaching/MTH541/Lecture%20notes/evaluation.pdf
Accessed 14 March 2014

14. Sokolovia Marina, Japkowicz Nathalie, Szpakowicz Stan. Beyond Accuracy, F-score and R OC: Family of Discriminant Measures for Performance Evaluation [Online]. Diro, University of Montreal, Montreal Canada;
URL:http://rali.iro.umontreal.ca/rali/sites/default/files/publis/EvAAAI06finMay10.pdf
Accessed 25 March 2014

15. Howard Hamilton. Computer Science 831: Knowledge Discovery in Databases, Confusion Matrix. 2012 Jun 8;

URL:http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html
Accessed 5 March 2014

16. Howard Hamilton. Computer Science 831: Knowledge Discovery in Databases, ROC Graph. 2012 Jun 8;
URL: http://www2.cs.uregina.ca/~dbd/cs831/notes/ROC/ROC.html
Accessed 5 March 2014

17. The Language of Technical Computing. [Online]. Math Works; 2014.
URL: http://www.mathworks.se/products/matlab/
Accessed 7 March 2014

18. Kotsiantis S.B, Kanellopoulos D. and Pintelas P. E. Data Preprocessing for Supervised Leaning. INTERNATIONAL JOURNAL OF COMPUTER SCIENCE [Online]. 2006; volume 1.
URL: http://www.math.upatras.gr/~esdlab/en/members/kotsiantis/v1-2-14.pdf
Accessed 10 March 2014

19. Hal Daumé III. A Course in Machine Learning [Online]. 2012.
URL: http://ciml.info/dl/v0_8/ciml-v0_8-all.pdf
Accessed 10 March 2014

20. Padraig Cunningham, Sarah Jane Delany. k -Nearest Neighbors Classifiers [Online]. Dublin Institute of Technology; 2007.
URL: http://www.csi.ucd.ie/UserFiles/publications/UCD-CSI-2007-4.pdf
Accessed 15 March 2014

21. Ian Witten, Eibe Frank. Data Mining: Practical Machine Learning Tools and Techniques, Second Edition. 2nd edition. San Diego; Los Angeles: Morgan Kaufmann; 2005.

22. Jeff Schilimmer. Congressional Voting Records Data Set [Online]. Washington Dc: Centre for Machine Learning and Intelligent Systems; 1987.
URL: http://archive.ics.uci.edu/ml/datasets/Congressional+Voting+Records
Accessed 15 January 2014

Appendixes

Appendix 1: Sample of row data

```
republican,n,y,n,y,y,y,n,n,n,y,?,y,y,y,n,y
republican,n,y,n,y,y,y,n,n,n,n,n,y,y,y,n,?
democrat,?,y,y,?,y,y,n,n,n,n,y,n,y,y,n,n
democrat,n,y,y,n,?,y,n,n,n,n,n,y,n,n,n,y
democrat,y,y,y,n,y,y,n,n,n,n,y,?,y,y,y,y
democrat,n,y,y,n,y,y,n,n,n,n,n,n,y,y,y,y
democrat,n,y,n,y,y,y,n,n,n,n,n,n,?,y,y,y
republican,n,y,n,y,y,y,n,n,n,n,n,n,y,y,?,y
republican,n,y,n,y,y,y,n,n,n,n,n,y,y,y,n,y
democrat,y,y,y,n,n,n,y,y,y,n,n,n,n,n,?,?
republican,n,y,n,y,y,n,n,n,n,n,?,?,y,y,n,n
republican,n,y,n,y,y,y,n,n,n,n,y,?,y,y,?,?
democrat,n,y,y,n,n,n,y,y,y,n,n,n,y,n,?,?
democrat,y,y,y,n,n,n,y,y,y,?,y,y,?,n,n,y,?
republican,n,y,n,y,y,y,n,n,n,n,n,y,?,?,n,?
republican,n,y,n,y,y,y,n,n,n,y,n,y,y,?,n,?
democrat,y,n,y,n,n,n,y,n,y,?,y,y,y,?,n,n,y
democrat,y,?,y,n,n,n,y,y,y,n,n,n,n,y,n,y,y
republican,n,y,n,y,y,y,n,n,n,n,n,?,y,y,n,n
democrat,y,y,y,n,n,n,y,y,y,n,y,n,n,n,y,y
democrat,y,y,y,n,n,?,y,y,n,n,y,n,n,n,y,y
democrat,y,y,y,n,n,n,y,y,y,n,n,n,?,?,y,y
democrat,y,?,y,n,n,n,y,y,y,n,n,?,n,n,y,y
democrat,y,y,y,n,n,n,y,y,y,n,n,n,n,n,y,y
democrat,y,n,y,n,n,n,y,y,y,n,n,n,n,n,y,?
democrat,y,n,y,n,n,n,y,y,y,y,n,n,n,n,y,y
democrat,y,n,y,n,n,n,y,y,y,n,y,n,n,n,y,y
democrat,y,y,y,n,n,n,y,y,y,n,y,n,n,n,y,y
republican,y,n,n,y,y,n,y,y,y,n,n,y,y,y,n,y
democrat,y,y,y,n,n,n,y,y,y,n,y,n,n,n,y,y
republican,n,y,n,y,y,y,n,n,n,n,n,y,y,y,n,n
democrat,y,y,y,n,n,n,n,y,y,y,n,n,n,n,y,?
democrat,y,y,y,n,n,n,n,y,y,y,y,n,n,n,y,y
republican,n,y,n,y,y,y,n,n,n,n,n,y,y,y,n,y
democrat,y,y,y,n,n,n,y,y,y,n,n,n,n,n,y,y
republican,n,y,n,y,y,y,n,n,n,n,n,y,y,y,n,n
republican,y,?,n,y,y,y,n,n,n,y,n,y,?,y,n,y
republican,y,y,n,y,y,y,n,n,n,n,n,n,y,y,n,y
republican,n,y,n,y,y,y,n,n,n,y,n,y,y,y,n,n
democrat,y,n,y,n,n,n,y,y,y,y,y,n,y,n,y,y
democrat,y,y,y,n,n,n,y,y,y,n,?,n,n,n,n,?
democrat,y,y,y,n,n,n,y,y,y,n,n,n,n,n,y,?
democrat,y,n,y,n,n,n,y,y,y,n,n,n,n,n,n,y
democrat,y,n,y,n,n,n,y,y,y,n,n,n,n,n,y,y
democrat,y,y,y,n,n,n,y,y,y,n,y,n,n,n,n,?
```

[22]

Appendix 2: Attribute Information

1. Class Name: 2 (democrat, republican)
2. Handicapped-infants: 2 (y,n)
3. Water-project-cost-sharing: 2 (y,n)
4. Adoption-of-the-budget-resolution: 2 (y,n)
5. Physician-fee-freeze: 2 (y,n)
6. El-Salvador-aid: 2 (y,n)
7. Religious-groups-in-schools: 2 (y,n)
8. Anti-satellite-test-ban: 2 (y,n)
9. Aid-to-Nicaraguan-contras: 2 (y,n)
10. Mx-missile: 2 (y,n)
11. Immigration: 2 (y,n)
12. Synfuels-corporation-cutback: 2 (y,n)
13. Education-spending: 2 (y,n)
14. Superfund-right-to-sue: 2 (y,n)
15. Crime: 2 (y,n)
16. Duty-free-exports: 2 (y,n)
17. Export-administration-act-south-Africa: 2 (y,n)[22]

Appendix 3: Sample of nominal conversion

```
1   Rep,5,10,5,10,10,10,5,5,5,10,0,10,10,10,5,10
2   Rep,5,10,5,10,10,10,5,5,5,5,5,10,10,10,5,0
3   Demo,0,10,10,0,10,10,5,5,5,5,10,5,10,10,5,5
4   Demo,5,10,10,5,0,10,5,5,5,5,10,5,10,5,5,10
5   Demo,10,10,10,5,10,10,5,5,5,5,10,0,10,10,10,10
6   Demo,5,10,10,5,10,10,5,5,5,5,5,5,10,10,10,10
7   Demo,5,10,5,10,10,10,5,5,5,5,5,5,0,10,10,10
8   Rep,5,10,5,10,10,10,5,5,5,5,5,5,10,10,0,10
9   Rep,5,10,5,10,10,10,5,5,5,5,5,10,10,10,5,10
10  Demo,10,10,10,5,5,5,10,10,10,5,5,5,5,5,0,0
11  Rep,5,10,5,10,10,5,5,5,5,5,0,0,10,10,5,5
12  Rep,5,10,5,10,10,10,5,5,5,5,10,0,10,10,0,0
13  Demo,5,10,10,5,5,5,10,10,10,5,5,5,10,5,0,0
14  Demo,10,10,10,5,5,10,10,10,0,10,10,0,5,5,10,0
15  Rep,5,10,5,10,10,10,5,5,5,5,5,10,0,0,5,0
16  Rep,5,10,5,10,10,10,5,5,5,10,5,10,10,0,5,0
17  Demo,10,5,10,5,5,10,5,10,0,10,10,10,0,5,5,10
18  Demo,10,0,10,5,5,5,10,10,10,5,5,5,10,5,10,10
19  Rep,5,10,5,10,10,10,5,5,5,5,5,0,10,10,5,5
20  Demo,10,10,10,5,5,5,10,10,10,5,10,5,5,5,10,10
    [illegible]
428 Rep,5,5,5,10,10,10,10,10,5,10,5,10,10,10,5,1
429 Demo,0,0,0,5,5,5,10,10,10,10,5,5,10,5,10,10
430 Demo,10,5,10,5,0,5,10,10,10,10,5,10,5,0,10,1 0
431 Rep,5,5,10,10,10,10,5,5,10,10,5,10,10,10,5,1 10
432 Demo,5,5,10,5,5,5,10,10,10,10,5,5,5,5,5,10 10
433 Rep,5,0,5,10,10,10,5,5,5,5,10,10,10,10,5,10 10
434 Rep,5,5,5,10,10,10,0,0,0,0,5,10,10,10,5,10
435 Rep,5,10,5,10,10,10,5,5,5,10,5,10,10,10,0,5 0
```